

METHOD AND APPARATUS FOR AVOIDANCE OF ROUTE-DESIGNATION DEADLOCK OF PACKET

Patent number: JP7282018
Publication date: 1995-10-27
Inventor: HARISHI SESU; ROBAATO FUREDERITSUKU SUTATSUK; KUREIGU BURAIA SUTANKERU
Applicant: IBM
Classification:
 - International: G06F15/16; G06F15/173
 - european: H04L12/56C
Application number: JP19950012797 19950130
Priority number(s): US19940222284 19940404

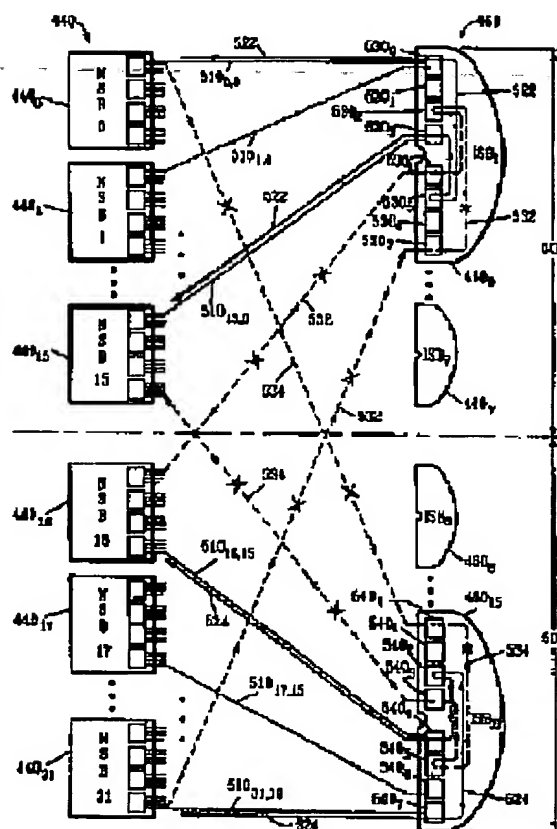
Also published as:

EP0676703 (A2)
 US5453978 (A1)
 EP0676703 (A3)
 EP0676703 (B1)

Report a data error here

Abstract of JP7282018

PURPOSE: To provide a device and method to establish path designation without a deadlock in a large scale 2-way multi-stage interconnecting cross point switch base packet network.
CONSTITUTION: In the case of selecting a path included in a path table in a system, the entire network is effectively separated so as to inhibit specific paths to separate the system into prescribed divisions, e.g. to separate a packet traffic mostly flowing among nodes of a half part 503 of the system from a packet traffic flowing among nodes of the other half part 507 in the system. In order to extract paths of a packet passing among nodes in a common division of the system from this standpoint, paths including paths 522, 524 passing through the other system division are inhibited. The path inhibit as above is not caused in the selection of paths through which a packet is propagated among nodes in a plurality of the system divisions, e.g. nodes included in different halves of the system.



Data supplied from the esp@cenet database - Worldwide

BEST AVAILABLE COPY

(19)日本国特許庁(JP)

(12)公開特許公報 (A)

(11)特許出願公開番号

特開平7-282018

(43)公開日 平成7年(1995)10月27日

(51)Int. Cl.⁶

G 0 6 F 15/16
15/173

識別記号

4 7 0 A

庁内整理番号

F I

技術表示箇所

G 0 6 F 15/16 4 0 0 N

審査請求 有 請求項の数 2 0 O L

(全 1 8 頁)

(21)出願番号 特願平7-12797

(22)出願日 平成7年(1995)1月30日

(31)優先権主張番号 222284

(32)優先日 1994年4月4日

(33)優先権主張国 米国 (US)

(71)出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)

(72)発明者 ハリシ・セス

アメリカ合衆国12401、ニューヨーク州キングストン、ナンバー310、ウィルバー・アベニュー 162-182

(74)代理人 弁理士 合田 潔 (外2名)

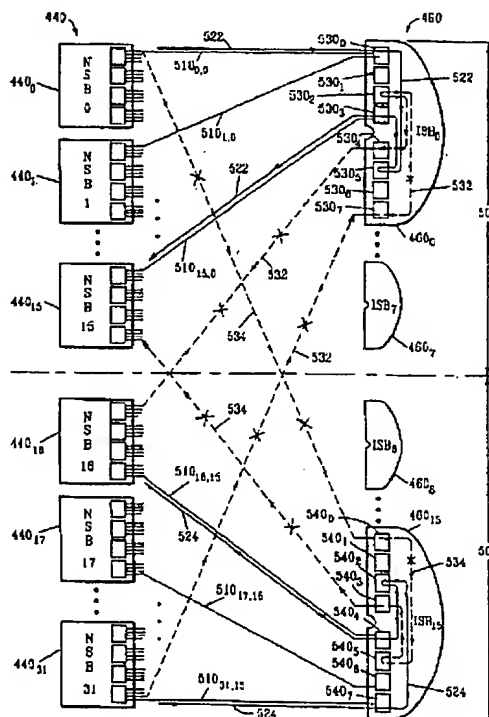
最終頁に続く

(54)【発明の名称】 パケットの経路指定デッドロック回避方法及び装置

(57)【要約】

【目的】 大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおけるデッドロックの無い経路指定を確立する装置及び方法を提供する。

【構成】 システム内の経路テーブルに含まれる経路の選択において、システムのある区分、例えばシステムの半分503内のノード間をもつぱら流れるパケット・トラフィックを、他の区分、例えばシステムの別の半分507内のノード間を流れるパケット・トラフィックから分離するために、特定の経路を禁止するように、ネットワーク全体が効果的に区分される。この点に関し、システムの共通区分内のノード間を通過するパケットの経路を抽出するために、他のシステム区分を通過するパス524、544を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分に含まれるノード間でパケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。



【特許請求の範囲】

【請求項 1】 バケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有する装置において、バケットが前記ネットワーク及び少なくとも 1 つの前記スイッチを介して、規定経路上を第 1 の前記ノードから第 2 の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する実質的な方法であって、バケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を第 1 に定義するステップであって、前記の各定義経路が少なくとも 1 つのリンクに伸び、第 1 のネットワーク区分だけに接続される第 1 及び第 2 の前記ノード間を通過するバケットが、第 2 のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第 1 及び前記第 2 のネットワーク区分に分割するように前記規定経路を定義する、前記第 1 の定義ステップと、全ての前記規定経路を結果の経路テーブルに記憶するステップと、を含む、方法。

【請求項 2】 前記第 1 の定義ステップが、前記第 1 及び前記第 2 のネットワーク区分にそれぞれ接続される第 3 及び第 4 のノード間を通過するバケットが、前記第 1 及び前記第 2 のネットワーク区分間に伸びる少なくとも 1 つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する第 2 の定義ステップを含む、請求項 1 記載の方法。

【請求項 3】 前記ネットワーク内において出所ノードから宛先ノードに経路指定されるバケットをアSEMBルする際に、前記第 3 及び前記第 4 の両ノード内において、前記バケットの結果的経路を生成するために、前記経路テーブルをアクセスするステップと、前記結果的経路を前記バケットにコピーするステップと、前記バケットを前記結果的経路上で前記ネットワークを介して経路指定するステップと、を含む、請求項 2 記載の方法。

【請求項 4】 前記経路テーブルの異なる部分を、前記複数の各ノードに対応する別々の局所経路テーブルにダウンロードするステップであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記ダウンロード・ステップを含み、前記経路コピー・ステップが、前記出所ノードから前記宛先ノードに伝搬されるバケットの前記結果的経路を生成するために、前記バケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセス

するステップを含む、請求項 3 記載の方法。

【請求項 5】 前記の各バケットが少なくとも 1 つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各バケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各バケットが対応する前記クロスポイント・スイッチの 1 つを横断する経路を定義し、前記結果的経路のコピー・ステップが前記結果的経路内の各連続する前記経路バイトの値を前記ヘッダ内の別々の対応する連続経路バイトにコピーするステップを含む、請求項 4 記載の方法。

【請求項 6】 各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項 5 記載の方法。

【請求項 7】 前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第 1 の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記バケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、請求項 5 記載の方法。

【請求項 8】 前記第 1 の定義ステップが、トポロジ・ファイル内のネットワーク装置及び相互接続データに応答して、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの 1 つへの全ての使用可能な最短パス経路を決定するステップであって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定ステップと、ある前記出所ノードとある前記宛先ノード間で 1 つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込むステップと、

ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する 1 つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択するステップと、前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分するステップと、を含む、請求項 5 記載の方法。

【請求項 9】 前記全ての使用可能な経路の決定ステップが、前記全ての使用可能な最短パス経路を突き止めるブレッズ・ファースト探索を実行するステップを含む、請求項 8 記載の方法。

【請求項 10】 前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第 1 の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記バ

ケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、請求項 9 記載の方法。

【請求項 1 1】各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項 1 0 記載の方法。

【請求項 1 2】パケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有するシステムにおいて、パケットが前記ネットワーク及び少なくとも 1 つの前記スイッチを介して、規定経路上を第 1 の前記ノードから第 2 の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する装置であって、パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を定義する第 1 の手段であって、前記の各定義経路が少なくとも 1 つのリンクに伸び、第 1 のネットワーク区分だけに接続される第 1 及び第 2 の前記ノード間を通過するパケットが、第 2 のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第 1 及び前記第 2 のネットワーク区分に分割するように前記規定経路を定義する、前記第 1 の定義手段と、全ての前記規定経路を結果の経路テーブルに記憶する手段と、を含む装置。

【請求項 1 3】前記第 1 の定義手段が、前記第 1 及び前記第 2 のネットワーク区分にそれぞれ接続される第 3 及び第 4 のノード間を通過するパケットが、前記第 1 及び前記第 2 のネットワーク区分間に伸びる少なくとも 1 つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する、請求項 1 2 記載の装置。

【請求項 1 4】前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする間に、前記第 3 及び前記第 4 の両ノード内において、前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスし、前記結果的経路を前記パケットにコピーし、前記パケットを前記結果的経路上で前記ネットワークを介して経路指定する手段を含む、請求項 1 3 記載の装置。

【請求項 1 5】前記経路テーブルの異なる部分がダウンロードされる前記複数の各ノードに対応する別々の局所経路テーブルであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記局所経路テーブルと、前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスする手段と、を含む、請求項 1 4 記載の装置。

【請求項 1 6】前記の各パケットが少なくとも 1 つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの 1 つを横断する経路を定義し、前記結果的経路内の各連続する前記経路バイトの値を、前記ヘッダ内の別々の対応する連続経路バイトにコピーする、請求項 1 5 記載の装置。

【請求項 1 7】各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項 1 6 記載の装置。

【請求項 1 8】前記第 1 の定義手段が、トポロジ・ファイル内のネットワーク装置及び相互接続データにตอบสนองして、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの 1 つへの全ての使用可能な最短パス経路を決定する手段であって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定手段と、

ある前記出所ノードとある前記宛先ノード間で 1 つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込む手段と、ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する 1 つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択する手段と、

前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分する手段と、を含む、請求項 1 6 記載の装置。

【請求項 1 9】前記システムが並列処理システムであり、前記の各ノードが別々の処理要素を含む、請求項 1 6 記載の装置。

【請求項 2 0】前記並列処理システムが 5 1 2 個の別々の処理要素を含み、前記スイッチが 3 2 ポート・スイッチ・ボードに編成され、前記システムが 3 2 個のノード・スイッチ・ボード (NSB) と 1 6 個の中間スイッチ・ボード (ISB) による複数のスイッチ・ボードを含み、前記 ISB が、前記の各 NSB 上の 1 6 ポートのそれぞれが、異なる対応するリンクを介して、前記の各 NSB 上の同一の対応するポートに接続されるように、また前記の各 NSB 上の残りの 1 6 ポートが 1 6 個の異なる連続する前記処理要素に接続されるように、全ての前記 NSB を集合的に相互接続する、請求項 1 9 記載の装置。

【発明の詳細な説明】

【0 0 0 1】

【産業上の利用分野】本発明はマルチステージ相互接続

クロスポイント・ベースのバケット交換を確立するための装置及び方法に関する。特に本発明は、大容量並列処理システム内で使用される高速バケット・ネットワーク内に組込むのに適するが、それに限るものではない。

【0002】

【従来の技術】強力で知能的で比較的安価なマイクロプロセッサの継続的な進歩及び市場での可用性により、大容量並列処理が、これまで従来式のメインフレーム・コンピュータにより処理されてきた広範なアプリケーション、例えばトランザクション処理、シミュレーション及び構造解析などを処理するために、益々魅力的な手段に成りつつある。

【0003】大容量並列処理システムでは、しばしば数百または数千にも上る相当数の比較的単純なマイクロプロセッサを基本とする別々の処理要素が、一般に高速バケット・ネットワークから形成される通信構造を介して相互接続され、各こうした処理要素がネットワーク上の別々のポートとして現れる。この構造はバケット形式の構造経路メッセージを、これらの処理要素の任意の1つから他へ経路指定し、それらの間の通信を提供する。これらの各々の処理要素は、通常、別々のマイクロプロセッサ及びその関連支援回路を含み、後者はとりわけ、一時記憶用のランダム・アクセス・メモリ(RAM)及び永久記憶用の読出し専用メモリ(ROM)、及び入出力回路により代表される。各処理要素は更に通信サブシステムを含み、これは適切な通信インタフェース及び他のハードウェア、並びにこの要素をバケット・ネットワークにインタフェースするように集散的に機能する制御ソフトウェアにより形成される。

【0004】一般に、大容量並列処理システムの全体性能は、そこで使用される根元的なバケット・ネットワークの性能により制限される。その点でバケット・ネットワークが余りに遅く、特に全体システム・スループットに悪影響を及ぼす程度に遅いと、結果的な低下は著しく、所与のアプリケーションにおいて大容量並列処理システムを使用する効果が低減する。

【0005】特に、大容量並列処理システムにおいて、各処理要素はアプリケーションの予め定められた細分化部分を実行する。その対応するアプリケーション部分の実行において、各要素は一般に、例えば異なる要素上で実行されるアプリケーション部分からデータを要求し、処理結果データを例えば、更に別の処理要素上で実行される別のアプリケーション部分に提供する。全ての要素間における処理の相互依存の性質により、各処理要素は、その時これらの各々の要素において実行されるアプリケーション部分からの要求により、データを別のこうした要素に転送できなければならない。一般に処理要素、例えば“宛先”要素が、別のこうした要素、例えば“出所”要素または“発信元”要素に対してデータを要求すると、宛先要素は少なくともこの特定のアプリケーション

ン部分に関し、その要素が出所要素により伝送される必要データを含むバケットを受信するまで遊休状態を維持する。バケットを受信すると、宛先要素は再度このアプリケーション部分の処理を開始する。バケット・ネットワークを通じて宛先からの要求を含むバケットを出所処理要素に移送し、次に要求データを含む応答バケットを反対方向に移送するためには、有限量の時間が必要である。この時間は、宛先要素において実行されるそのアプリケーション部分に、ある程度の待ち時間を不可避に挿入する。システム内のほとんどの処理要素が、出所要素において実行されるアプリケーション部分に対応する宛先要素として機能するので、この通信により誘導される待ち時間が余りに長いと、システム・スループットが顕著に低下する。結果的に、このことは全体システム性能を著しく低下させることになる。これを回避するためにバケット・ネットワークは各バケットを任意の2つの通信処理要素間で、この待ち時間を低減するように、可能な限り速く移送しなければならない。更に、典型的な大容量並列処理システムにおいて一般に使用される相当数の処理要素、及びこのシステム内の任意のある要素が、任意の時刻において、他のこうした要素と通信するために必要な付随のニーズを考慮すると、ネットワークは相当地に大きな数、例えば予測されるピーク負荷のバケットを処理要素間で同時に経路指定できなければならない。

【0006】しかしながら、実際には、大容量並列処理システムで使用される必要性能、特に伝送帯域幅を所有するバケット交換ネットワークは、様々な理由から、その開発が極めて困難であることがわかっており、そのためにこうしたシステムの急速な普及及び使用の増加がある程度阻止されてきた。

【0007】バケット・ネットワークの様々な形態が今日存在するが、1つの共通のアーキテクチャとしては、比較的小さなクロスポイント・スイッチのマルチステージ相互接続構成を使用する。各スイッチは、通常、8ポート双方向ルータであり、全てのポートがクロスポイント・マトリックスを通じて内部的に相互接続される。こうしたネットワークでは、1ステージ内の各スイッチはネットワークの片側(すなわち、いわゆる“入力”)において開始し、特定の対応するパス(典型的にはバイト幅物理接続)を通じて、次の続くステージ内のスイッチに相互接続され、このようにして、ネットワークの反対側(すなわち、いわゆる“出力”)の最後のステージに到達するまで継続される。こうしたスイッチは、今日、動作的には非ブロッキング(non-blocking)の比較的安価な単一の集積回路(以降では“スイッチ・チップ”として参照される)として調達可能であるので、これらのスイッチ・チップが好まれて使用される。実際に、中央キューの使用に頼る非ブロッキング8ウェイ・ルータとして実現されるこうしたスイッチ・チップが、P. Hochschildらによる係属中の米国特許出願第027906号“A Cen

tral Shared QueueBased Time Multiplexed Packet Switch with Deadlock Avoidance”(1993年3月4日出願)に述べられている(本願の出願人に権利譲渡される)。

【0008】こうした双方向マルチステージ・パケット交換ネットワークは、他のパケット交換ネットワーク・トポロジと比較して比較的単純であり、その全てのポート間で高い伝送帯域幅を提供するが、残念ながらこのタイプのネットワークは、経路指定デッドロックを受け易い。これらのデッドロックは稀にしか発生しないが、同一ステージ内の任意の2つのスイッチ間に複数の経路が存在するために、実際に発生する。

【0009】この点に関し、8個のこうしたスイッチ・チップが2つの相互接続ステージに編成される単純な32ポート・ネットワークについて考えてみよう。すなわち、4個のスイッチによる入力ステージの後に、4個のスイッチによる出力ステージが続き、これらの全てのスイッチ・チップが単一のスイッチ・ボード上に含まれる。この構成では、入力ステージにおいて、異なるスイッチ・チップ上の任意の2つのポート間を通過するパケットは、出所(“入力”)ポートを含む入力ステージ内のスイッチ・チップを通過して、出力ステージの4個のスイッチ・チップの1個に経路指定される。次に、この後者のスイッチ・チップが、パケットをこのパケットの宛先(“出力”)ポートを含む入力ステージ内のスイッチに逆経路指定する(すなわち、その方向を反転する)。スイッチ・チップ間の経路は、通常、比較的短い時間に渡り、各バイト・ワイス(byte-wise)・パスがほぼ等しい数のパケットを伝搬し、ネットワーク全体を通じてのトラフィック・フローを平均化するように、システム初期化の間に予め定義される。これらの経路が設定されると、スイッチ・チップまたはバス故障或いは保守状態以外では経路は稀にしか変更されない。各処理要素が使用可能な割当てられた経路が、次に再度システム初期化の間に(局所)経路テーブルの形式でその要素に提供される。引続きルーチンのオペレーションの間に、各処理要素がパケットを形成すると、その要素はこのパケットの宛先にもとづきその経路テーブルから経路を読み出し、単にその経路をパケットのヘッダ内に適切な経路バイトの値として挿入する。パケットが次にネットワーク内に送出され、パケット内の対応する経路バイトの値により指定される継続するスイッチ・チップ(及び交換ステージ)を経由して、経路指定される。パケットが交換ステージを経由して横断すると(すなわち、ここでは同一ステージの2個のスイッチ・チップを通過する)、ステージ内の最後のスイッチ・チップがパケット・ヘッダから対応する経路バイトを切捨てる。

【0010】経路は従来、経路指定デッドロックの潜在性を考慮すること無く定義されてきた。従って、各々が例えば異なるスイッチ・チップのグループの中央キュー

内に存在する対応するパケットが、連続ステージ内のスイッチ・チップ対を接続する共通バス上を同時に経路指定されるのを待機する度に、経路指定デッドロックが発生する。こうした状態が発生すると、これらの各々のスイッチ・チップは、グループ内の他のスイッチ・チップがそれらのパケットをこれらの特定のバス上に経路指定するのを待機する。このグループのどのパケットも、このグループの任意の1つのパケットが経路指定されるまで、その関連する中央キューを通過することができないので、これら全てのパケットがひたすら待機し、対応するバスがデッドロック状態となり、その上をトラフィック・フローが生じなくなる。その結果、デッドロックが発生すると、これらのパケットが宛先指定される処理要素についても、これらのパケットを待機し続けることになり、それらの処理のスループットを停止させる。結果的にネットワークの帯域幅はデッドロックにより影響されない残りの処理要素だけを優遇するようになり、処理の作業負荷が著しく偏り、システム・スループットを多大に低下させることになる。

【0011】デッドロックを回避する問題に直面して、当業者は最初に経路指定デッドロックを予測するために、特定のタイプの大域アービトレーション手法が使用可能であると考え、多数の非デッドロック状態のバスのいずれかを選択し、その上でパケットを送信し、デッドロックを回避することを期待するであろう。この手法は、潜在的な経路指定デッドロックを検出し、それに従い調停するために全ての中央キューを通過する全てのパケットがモニタされることを必要とする。残念ながら、これらの機能を達成する回路は極めて複雑であり、全ての各スイッチ回路の外部に配置されて、それらの各々と接続される必要がある。これはパケット交換ネットワークのサイズ、複雑度、従ってコストを押し上げることになる。この手法自体は、極めて非現実的である。

【0012】こうしたことを考慮して、当業者は2重のスイッチ・ボードを有するパケット・ネットワークを形成するなどの別の手法に注目するであろう。この手法を32プロセッサ・システムと共に使用することにより、1つのスイッチ・ボードのポート16乃至31で表される16個のポートが、別のスイッチ・ボードの同じポートに接続される。両方のボード上の残りの各ポート0乃至15は、32個の別々の処理要素の対応する1つに接続される。オペレーションにおいて、共通のスイッチ・ボードに接続される出所ポートと宛先ポートとの間を通過するパケットが、もっぱらその1つのスイッチ・ボード内に経路指定され、他のスイッチ・ボード内に含まれるスイッチ・チップに影響を及ぼすことはない。異なるスイッチ・ボード上の出所ポートと宛先ポートとの間を経路指定されるパケットだけがボード間を経路指定される。片方のスイッチ・ボード内だけを流れるパケットと他のスイッチ・ボード内だけを同時に流れるパケットと

潜在的に相互作用しないように分離することにより、この手法はデッドロックを排除する。更にこの手法は伝送帯域幅を悪化させない。残念なことにこの手法は2重のスイッチ・ボード及び関連回路を必要とすることにより高価である。それにも関わらず、スイッチ・ボード及び関連回路を2重化する追加のコストが32プロセッサ・システムにおいては許容可能である。この手法自体が、32プロセッサ・システムにおけるデッドロックを回避するために使用される。実際に32プロセッサ・システムでは、1つのスイッチ・ボードだけではパケット・ネットワークの形成を妨げる十分なデッドロックの潜在性が存在する。しかしながら、このコスト的な欠点は、例えば512プロセッサ・システムなどのように、ネットワーク内で必要とされる最小16個のスイッチボードに加え、追加の16個のスイッチ・ボードを必要とする大規模システムの場合に、高額でより付けられない過ぎない。

【0013】最後に、当業者は、特定の経路の使用を単に禁止することにより、経路指定デッドロックを回避する手法を考慮するであろう。この特定の手法により、同一ステージ内の2個のスイッチ・チップ間の全ての経路の特定のサブセットだけが、それらの間のパケット・トラフィックの伝搬に使用可能と定義され、経路テーブル内に含まれる。1度選択されると、これらの経路は保守状態または故障状態以外では変化しない。サブセットを形成する経路は、特に経路指定デッドロックが発生しないように選択される。各追加の経路が禁止されるとネットワーク帯域幅が低下するので、この手法の目標はできる限り少ない経路を禁止することである。

【0014】しかし残念ながら、経路が禁止されると”禁止されない”経路がシステム内の全てのノードに関して、対称でないことが知られている。その結果、伝送帯域幅がネットワーク全体に渡り均等に低減されず、ネットワーク全体に渡って帯域幅の非対称が生じる。これらの非対称の結果、ネットワークは伝送帯域幅が特定の”ホット”・ポートにおいて非常に高くなる傾向にあり、他では実質的に0となる、いわゆる”ホット・スポット”を発展させる傾向を示す。これは次に、他のポートを犠牲にして”ホット”・ポートに関連する処理要素を優先するように処理スループットを偏らせ、ネットワーク全体に渡る作業負荷処理の平衡を失わせる。結果的にシステム性能の低下が生じる。実際に、経路がもっぱらスイッチ・ボード内で禁止されると、ネットワーク全体に渡り一定の帯域幅の低下をもたらす残りの禁止されない経路の任意の組み合わせを見い出すことができないことが判明した。

【0015】経路を禁止する手法は、単に各処理要素に対応する経路テーブル内に含むための特定のエントリの選択を要求するだけなので、この手法は、具体化が非常に単純で高度にコスト有効である。従ってこの手法は、

ネットワーク全体に渡り対称的な帯域幅の低下を生成不能でない限り、マルチステージ・クロスポイント・パケット・ネットワークに取り入れられることが望まれる。

【0016】相互接続双方向マルチステージ・クロスポイント・ベースのネットワークを、大容量並列処理システムの通信中枢として使用する関心にも関わらず、これらのネットワークにおけるデッドロックの潜在性の増加、並びに特に大規模ネットワークにおける現実的なソリューションの欠如が少なくとも今日まで、32をはるかに越えるプロセッサを有するこうしたネットワークを使用する大容量並列処理システムの市場での使用可能性を抑制してきており、特定の大規模処理アプリケーションにおけるこれらのシステムの使用を妨げてきた。

【0017】従って、大規模双方向マルチステージ相互接続クロスポイント交換ネットワークにおいて、特に、大規模大容量並列処理システムにおいて、デッドロックの発生を防止する現実的な手法が必要とされる。こうした手法は具体化が単純であり、高度にコスト有効であるべきであり、ネットワーク帯域幅が結果的に低減される場合、ネットワーク全体に渡り、実質的に対称で受諾可能なレベルの帯域幅の低減を提供するべきである。こうした手法がこうしたシステム内に含まれると、これらのシステムは市販されると32をはるかに上回る、例えば512或いはそれ以上の別々のプロセッサに拡張される。従って、こうしたシステムは従来不可能であった追加のアプリケーション処理のニーズに応えることができる。

【0018】

【発明が解決しようとする課題】本発明により、大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおいて、経路指定デッドロックの発生を防止するための、従来技術に固有の欠点を有利に克服する単純でコスト有効な手法が提供される。この手法は、理想的には大規模大容量並列処理システムの通信中枢を形成するこうしたパケット・ネットワークにおいて使用される。

【0019】

【課題を解決するための手段】特に本発明により、特定の予め定義された経路が経路テーブルの形成の間に、これらのノードを使用するネットワーク内における特定のノード、例えば処理要素の相対ロケーションにもとづき考慮から除外される。禁止された経路は、もしそれらが使用されなければ、閉ループ経路指定パターン、従って経路指定デッドロックの発生を防止する経路として選択される。経路テーブルに含まれる経路の選択においてシステムのある区分、例えばシステムの半分内のノード間をもっぱら流れるパケット・トラフィックを別の区分、例えばシステムの他の半分内のノード間を流れるパケット・トラフィックから分離するために経路が禁止される。この点に関し、システムの共通区分内のノード間を

通過するバケットの経路を抽出するために、システムの別の区分を通過するパス（ケーブルなど）を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分に含まれるノード間でバケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。

【0020】例えば、 8×8 のスイッチ回路（ここでは”スイッチ・チップ”としても参照される）の使用において、多数の同一の320ポート・スイッチ・ボードを有する512プロセッサ・システムが構成され、これらが2つの相互接続ステージ、すなわち、個々の処理要素に接続されるノード・スイッチ・ボード（NSB）及びノード・スイッチ・ボード自身を相互接続するために使用される中間スイッチ・ボード（ISB）に編成される。各NSBは16個のそれぞれ異なる処理要素に接続される16ポートと、16個の各ISB上の異なるポートに相互接続される別の16ポートを提供する。

【0021】このシステムにおいて禁止される経路を決定するために、16個の連続的なNSB（例えばNSB0乃至15、及び16乃至31）及び256個の連続的な処理要素が各半分を構成するように、システムが半分に分割されて効果的に考慮される。第1の8個のISBが一方の半分に含まれ、残りの8個のISBが他の半分に含まれる。システムの共通の半分に配置される処理要素間を通過するバケットに対して、システムのその半分に完全に含まれるISBポートを含む使用可能経路だけが許可され他の経路は禁止される。従って、システム初期化の間に、後者の任意の経路は、これらの処理要素を接続する大域経路テーブル内に含まれない。或いはシステムの異なる半分に配置される処理ノード間を通過するバケットに対しては、こうした経路は禁止されない。従ってこの場合には、大域経路テーブル内に結果的に含まれる経路の選択は、システムの半分にもとづく制限無しに、使用可能な全ての経路の中から実施される。

【0022】システムの各区分例えば半分を、他の任意の区分例えば他の半分に含まれる処理要素対間をもつばら流れるバケット・トラフィックから分離することにより、これらのバケットの相互作用により生じる経路指定デッドロックが有利に防止される。これにより市販の並列処理システムは、より多くの処理要素を含むように容易に拡張され、従来可能であった以上に広範な様々なアプリケーション処理ニーズに応えることができる。

【0023】

【実施例】当業者には容易に理解されるように、双方向マルチステージ相互接続クロスポイント・ベース・バケット・スイッチを含むバケット・ネットワークは、それらの特定のアプリケーションに関係無く、ここで指摘されるタイプの経路指定デッドロックの影響を受け易い。従って次の説明を考慮した後に、当業者においては、本発明の教示がほとんどのこうしたバケット・ネットワークに容易且つ高度にコスト有効に組込まれ、これらのデ

ッドロックの発生を伝送帯域幅の僅かな低減により、防止することが理解されよう。従って、本発明は実質的に任意のサイズのバケット・ネットワークにおいて即刻使用され、公衆または専用電話回線（例えば局所、広域または首都圏ネットワーク）または他の類似のネットワークなどのデジタル通信、或いは大容量並列処理システムの通信中枢などの特殊アプリケーションに関わり無く、広範且つ様々な範囲のバケット交換環境に渡って使用される。しかしながら、後述の説明を単純化するために、本発明は大容量並列処理システム、そして特に、IBMにより今日製造されるスケーラブル並列処理システムのSPファミリにおいて使用されるIBM9076 SP-1高性能通信ネットワークにおいて使用されるように述べられる。

【0024】本発明の理解を容易にするために、最初に並列処理システムのバケット経路指定の様々な態様、特に、そこで使用される双方向クロスポイント方式バケット・ネットワークに関する態様について述べ、次に典型的な経路指定デッドロック状況について、そして最後に、これらのデッドロックの発生を有利に防止する本発明について詳細に述べることにする。

【0025】最初に、図1に示される従来の32プロセッサ並列処理システム5について考えてみる。このシステムは32個のノード・バケット・スイッチ100（ここでは”バケット・ネットワーク”または単に”ネットワーク”としても参照される）を含み、各ノードには32個の別々の（しかしながら一般には同一の）処理要素110（特に処理要素110₀、110₁、...、110₃₁）が接続される。各要素はシステムの処理ノードを形成する。ネットワークはこれらの処理ノードの1つから他のノードへの高速伝送を提供する。処理要素自身はそれぞれマイクロプロセッサを基本とし、通常、IBMにより製造されるRS6000 RISCマイクロプロセッサを使用する。本発明は任意のこれらの要素のアーキテクチャまたは回路には無関係であるので、当業者には容易に明らかとなろうこれらの態様については詳細には述べない。しかしながら、本発明は後に詳述されるように、これらの処理要素の1つにおいて実行されるシステム初期化ソフトウェア、及びこれらの各々の要素内に記憶される経路テーブル内において実現される。従って、これらの特定の態様については、特に後述される。

【0026】図示のように、ネットワーク100は8個の別々の 8×8 双方向スイッチ回路120により構成され、これらは2つの相互接続ステージ、すなわち4個のスイッチ回路120₀、120₁、120₂及び120₃を含む”入力”ステージと、4個のスイッチ回路120₄、120₅、120₆及び120₇を含む”出力”ステージとに編成される。”入力”及び”出力”の指定は、純粋に説明の都合において任意であり、実際には、ネットワーク上のステージまたはポートは入力または出力ステージ或いはポートとして機能する。これらの各々のスイッチ回路

は、好適には、中央キュー・ベースの非ブロッキング 8 ウェイ・ルータである。各スイッチ回路は単一の集積回路として、すなわち、いわゆる”チップ”として集積化され、ここでは各こうしたスイッチ回路自身を”スイッチ・チップ”として参照する。もちろん当業者には理解されるように、各スイッチ回路は単一のチップとしてだけ具体化される必要はない。いずれの場合にも、スイッチ・チップ自身は本発明の 1 部を形成しないので、これについては詳細には述べないことにして、この回路のその他の詳細に関して述べることにする。図示のように、各スイッチ・チップは中央キューを含み、これら是对応するスイッチ回路 120_a、120_b、120_c、...、120_n内のキュー 130_a、130_b、130_c、...、130_nとして表される。基本的に各中央キューの目的は、とりわけ入力阻止及びデッドロックを改良するために、対応するスイッチ回路を通過する別の経路を提供することであり、後者すなわちデッドロックは、入力ポート（特に内部の F I F O バッファ）及び逆のトラフィックにより充填されたキューに起因する（これは本発明が対象とするのとは異なる形態のデッドロックである）。

【0027】ネットワークの入力及び出力ステージは、接続マトリックス 140 を介して相互接続され、これらの各々の接続は実質的にバイト幅物理リンク（ケーブル）であり、特にそれらの内のリンク 140_a、140_b、140_c及び 140_nが番号付けされて示される。このマトリックスを介して、入力ステージ内のそれぞれのスイッチ・チップのポートが別々にまた物理的に出力ステージ内のあらゆるスイッチ・チップの対応するポートに接続される。例えば、スイッチ・チップ 120_aはポート 0 乃至 7 を備え、そのポート 4 乃至 7 を通じ、対応するケーブルを介して、各スイッチ・チップ 120_a、120_b、120_c及び 120_n上のポート 4 に接続される。8 個のスイッチ・チップ及び接続マトリックス 140 を含むパケット・スイッチ 100 は、集合的に単一のスイッチ・ボードを含む。各スイッチ・チップのポート 0 乃至 3 はスイッチ・ボード外のリンクに接続され、各スイッチ・チップのポート 4 乃至 7 は、接続マトリックス 140 内のリンク（ケーブル）に接続され、それを介して同一ボード内の別のスイッチ・チップのポートに接続される。

【0028】ある要素が別の要素からデータを要求したり、データを供給したりするなど処理要素が互いに通信するために、”出所”処理要素は自身が実行するアプリケーション部分にもとづき、命令またはデータと共に適切なメッセージを含むパケットを形成し、そのパケットを”宛先”処理要素に伝送するためにパケット・スイッチ 100 に送信する。宛先要素はパケット内に含まれるデータまたは命令を処理し、適切な応答を生成する。応答は次に、宛先処理要素において実行されるアプリケーション

ョン部分にもとづき、別のパケットに形成され、例えば出所または異なる処理要素に伝送して処理するためにネットワークに返送される。

【0029】ネットワークを介するパケット伝送を容易にするために、各パケットは経路バイト形式の特定の経路指定命令を有するヘッダを含む。後述のように、全ての経路が予め定義される。出所処理要素がアSEMBL 中の任意のパケットの宛先を決定すると、その要素は単に、宛先処理要素をアドレスとして有するその内部（局所）経路テーブルをアクセスし、適切な経路バイト値の形式で経路を読み出す。この値が単に経路バイトとしてパケットのヘッダに挿入される。

【0030】図 2 はパケット・ネットワークを通じて伝送される典型的なパケット、すなわちパケット 200 の構成を示す。個々のパケットは例えば 255 バイト長である。図示のように、パケット 200 は連続するフィールド、すなわち長さフィールド 210、経路フィールド 220（それ自身経路バイト 220_a、220_b、...、220_nを含む）、シーケンス番号フィールド 230 及びデータ・フィールド 240 を含む。長さフィールド 210 は、パケット長をバイトで指定する 8 ビット・ボリュームを含む。経路フィールド 220 は複数のバイト、特に経路バイト 220_a、220_b、...、220_nを含み、これらは集合的にパケットがネットワーク全体を通じてその出所ノードから宛先ノードに至る特定の単一の経路（パス）を指定する。フィールド 230 は出所処理要素により提供されるシーケンス番号を保持する。この番号は、このパケットに対応して出所処理要素により割当てられ、宛先処理要素により使用され、所与のシーケンスにおけるパケットの順番を識別する。この番号自体は、宛先におけるシーケンス外のパケットの処理の防止のためのチェックに使用される。データ・フィールド 240 は連続するバイトを含み、これらは集合的にパケットにより宛先処理ノードに伝搬されるデータ（実際のデータまたは命令を含む）を形成する。フィールド 210、220 及び 230 は集合的にパケット・ヘッダを形成する。

【0031】経路指定フィールド 220 に現れる経路バイトの数（n）は、パケットが通過する交換ステージの数により決定される。その点に関し、各経路バイトは 2 つの連続スイッチ・チップに対応する経路指定命令を保持する。従って、パケットが宛先処理ノードに達するまでに、図 1 に示されるように、ネットワーク内の 2 つの連続ステージ内の 2 個のスイッチ・チップを通過するだけであれば、フィールド 220 は経路バイト 220_aだけを含むことになる。レイヤ・ネットワーク（layer network）においては、追加の対のスイッチ・チップが使用される。全ての経路バイトが同一の形式を有する。この点に関し、経路バイト（R [7:0]）は 1 ビットのフィールド選択子（R [7]、図示せず）、及び 2 つの

3ビットの経路フィールド(R[6:4]及びR[2:0]、両者共に図示せず)を含む。ビットR[7]の値が0の場合、スイッチ・チップはパケットを2進値R[6:4]により指定されるそのチップ上の出力ポートに経路指定し、次にビットR[7]の値を1に設定する。或いはビットR[7]の値が1の場合、スイッチ・チップはパケットをビットR[2:0]で指定されるそのチップ上の出力ポートに経路指定し、その間に、この完全な経路バイトを廃棄する。このようにして、パケットから経路バイトを解析する。従って、各経路バイトは2つの連続スイッチ・チップに対する経路指定命令を提供する。n個の経路バイトを経路フィールド220内に連結することにより、各パケットはスイッチ・チップの最大2nのステージを通じて経路指定される。

【0032】要約すると、パケットを受信するスイッチ・チップは、そのパケット内にその時、存在する第1の経路バイトを調査し、そのパケットをそのバイトにより示されるポートに経路指定する。そうする間に、そのパケットのパス内のあらゆる別のスイッチ・チップは、その完全な経路バイトをパケットから切り取る(除去する)。これは次に、経路フィールド220内の次に続く経路バイトを、次のスイッチ・チップ及び交換ステージに対応する第1の経路バイトとして形成する。宛先処理ノードに到来する時、パケットは経路バイトを含んでいない。各スイッチ・チップはその時パケットにより伝搬される第1バイト以後の追加の経路バイトを意識せず、第1バイトに対してその回路は、その特定の経路指定を実行する。更に各スイッチ・チップは第1バイト以外の経路バイトと、続くデータ・バイトとを区別しない。

【0033】上述のように、経路指定はパケット・アセンブリの間に最初に予め定義された経路バイトをパケット・ヘッダに挿入し、次にそのパケットの実際の経路指定が導かれ指令されることにより、出所処理要素及び宛先処理要素に関係なく、これらの各々のバイトの特定の値によりネットワーク内において達成される。

【0034】図3は、図1に示されるシステム5を構成する処理ノード110を示し、特に、これらのノードのメモリ内に存在してパケット経路指定を実行する様々なファイル及びテーブルを示す。パケット・スイッチ(ネットワーク)100は時分割の2つのモードで機能する。それらの一方は実行フェーズであり、この間、スイッチ回路は単に到来パケットを経路指定する。他はサービス・フェーズであり、この間、プロセッサは初期化されるか、ネットワークが回線交換方式でモニタ及び管理される。ネットワークに接続される全てのスイッチが、モード間で同期化ロック・ステップ方式で転送する。実行フェーズの間、特定の処理要素は特定のタスクを任せられる。例えば、処理要素110₂及び110₃は、システム5から他のネットワークへの、または処理システムへのリンクを提供し、それらの間で情報を転送するため

の入出力ノードとして指定される。他の処理要素、例えば処理要素110₂、110₃、...、110₃₁は、全て実際のアプリケーション処理のための計算ノードとして使用される。処理要素の1つ、例えば処理要素110₃₁は、サービス・フェーズの間の様々なネットワークオペレーションを引受けるサービス・プロセッサとして使用される。必要に応じて実行フェーズの間、サービス・プロセッサは計算ノードとしても機能することができる。サービス・プロセッサはハードウェアの見地からは、他の全ての処理要素と同一であるが、サービス・プロセッサはそのメモリ(ここではメモリ340)内に、サービス・フェーズの間に実行される追加のソフトウェア、とりわけ初期化ルーチン370を含み、これを実行する。例えばこのフェーズは全てのスイッチ回路及びネットワークに接続される全ての他の装置(全ての他の処理要素を含む)に対して、初期化、通信リンク同期、大域時間同期、故障判断、及び分離、及び様々な診断サービスを提供する。初期化機能はサービス・フェーズの1部に過ぎないのでサービス・フェーズのこの部分、特にパケット経路指定及び本発明に関連する観点についてのみ、以降で述べることにする。初期化フェーズは、システムが任意のアプリケーション処理を請け負う以前に、請け負われる。

【0035】サービス・プロセッサ110₃₁はそのメモリ340内に、ネットワークに接続される全ての処理要素を含むそれぞれの及びあらゆる装置、及びこれらの装置をリンクするためにネットワーク内において使用される特定の双方向物理接続(ケーブル)を集散的に定義する構造化エントリのデータベース、特にトポロジ・ファイル350を記憶する。データベースが生成される方法は本発明には関連しないので、ここでは触れないことにする。トポロジ・ファイルにおいて、スイッチ回路及び他の装置の最大数が装置エントリにより最初に識別され、任意のこれらの回路及び装置間に存在する各物理接続のエントリがそれに続く。装置エントリは2つの数値フィールドを含み、これらは"装置番号(n_v)；スイッチ回路番号(n_s)"の形式を取る。これらの値が提供されると装置識別(id)の番号付けが0乃至n_vの範囲において、またスイッチ回路idの番号付けが0乃至n_sの範囲において仮定される。最大16個の装置及び8個のスイッチ回路を含むネットワークでは、装置エントリは単に"168"である。各接続エントリは6つのフィールドを有し、これは"装置1タイプ；装置1id；装置1ポート；装置2タイプ；装置2id；装置2ポート"の形式を取る。装置タイプ情報は装置の性質、すなわちその装置が処理要素かどうかを指定し、そうであれば、その要素がサービス・プロセッサかどうか、或いはスイッチ回路かどうかを指定する。接続エントリの例は"tb0140s36"であり、これは"14の処理要素が全2重方式で、そのポート0からスイッチ回路3の入出力

両ポート6に接続される”ことを意味する。ネットワークの配線は、通常、極めて規則的であり、良好に定義され対称的である。しかしながら、実際には幾つかのスイッチ・ボードは、保守状態或いは故障状態の結果として、故意に分離される他のネットワーク・コンポーネント、例えばケーブル、スイッチ回路（特に使用されるスイッチ・チップ）または処理要素のために、パワー・ダウン状態の可能性がある。従って、任意の瞬間におけるネットワーク・トポロジは極めて不規則であったりする。

【0036】 いずれにしても、初期化及び特に初期化ルーチン370の実行の間、サービス・プロセッサ110₃₃はその時存在するトポロジ・ファイル350を読み出し、次にテスト・メッセージを同報し、それに対応する応答を受信することにより、ネットワークに接続される各装置と同様、ネットワーク内の各接続の状態を物理的に判断する。これらの応答にもとづき、サービス・プロセッサは、例えば既知のブレッズ・ファースト探索(breadth-first search)により、ネットワークの各(出所)ノードをネットワークのあらゆる他の(宛先)ノードに接続するための全ての使用可能な経路を判断する。双方向マルチステージ・クロスポイントネットワークに固有のバス冗長性により、異なるスイッチ・ステージ内の異なるスイッチ回路を通過して、1対の出所ノード及び宛先ノードを接続する複数の経路がしばしば存在する。各共通の出所/宛先ノード対間の複数の経路を鑑み、サービス・プロセッサは次にこれらの各々のノード対に対応するこれらの経路の1つを選択し、その経路をメモリ340内の大域経路テーブル360に記憶する。これらの経路はネットワーク内におけるトラフィック渋滞及びホット・スポットを回避するために単位時間に渡り、ネットワーク全体を通じてパケット・トラフィックの実質的に一様な分布を達成するように、主に最短パスにもとづき選択される。

【0037】ネットワーク100の使用可能な各出所／宛先ノード対間のパスを定義する大域経路テーブル360が完全に構成されると、サービス・プロセッサ110₃₁は次にネットワークを通じ、そのテーブルの対応部分を自身を含む各個々の処理要素に局所経路テーブルとして、そこに記憶するために提供する。この部分は、その特定の処理要素を出所ノードとしてリストする経路だけを含む。従って、例えば処理要素110₀はそのメモリ310内に、局所経路テーブル320を記憶し、サービス・プロセッサ110₃₁はそのメモリ340内に、局所経路テーブル380を記憶する。他の処理要素についても同様である。パケット形成の間、上述のように、各処理要素は単にその局所経路テーブルをアクセスするだけで、その時アセンブルされるパケットの宛先にもとづき、その宛先の経路指定バイトの値をテーブルからそのパケットのヘッダにコピーする。

【0038】上述の説明を考慮して、経路指定デッドロックを表す図1を再度参照することにする。

【0039】経路指定デッドロックは、各々が例えばスイッチ・チップの異なる交換ステージ内の中央キューに存在する対応パケットが、連続するステージ内のスイッチ・チップ対を接続する共通バス上における経路指定を同時に待機する度に発生する。従って、ここで“A”と記されるパケットがスイッチ・チップ120の中央キュー130に内在し、処理ノード110から“丸A”で示される破線のバスを介して、処理ノード110₄に経路指定されるのを待機しているものと仮定する。このバスを通じ、パケット“A”はスイッチ・チップ120により、ケーブル140を介してスイッチ・チップ120₄のポート4に導かれ、次にこの後者のチップのポート5及びケーブル140₁を介して、入力ステージ特に処理ノード110₄に接続されるスイッチ・チップ120₁のポート0に経路指定されて戻される。同様にキュー130に内在するパケット“A”と同時に、スイッチ・チップ120₄、120₁及び120₅のそれぞれの中央キュー130₄、130₁及び130₅に、3つの他のパケット“B”、“C”及び“D”が内在するものと仮定する。パケット“B”はスイッチ・チップ120₄のノード1に接続される処理要素110₁₇から、“丸B”で示される破線のバスを介して、スイッチ・チップ120₅のノード3に接続される処理要素110₂₁に経路指定される。同様にパケット“C”は、スイッチ・チップ120₁のノード2に接続される処理要素110₆から、“丸C”で示される破線のバスを介して、スイッチ・チップ120のノード2に接続される処理要素110₂に経路指定される。同様にパケット“D”は、スイッチ・チップ120₅のノード1に接続される処理要素110₂₁から、“丸D”で示される破線のバスを介して、スイッチ・チップ120₄のノード0に接続される処理要素110₁₆に経路指定される。

【0040】図示のように、全ての4つのバケットは同時に衝突する経路を有し、同一セットの4つのケーブルを介する。各経路はそのケーブルを他の2つの経路と共用する。結果的に、各スイッチ・チップ120_a、120_b、120_c及び120_dは、対応する中央キューに内在するこれらのバケットと共に、これらのスイッチ・チップの任意の他の1つが最初にそのバケットを経路指定するのを待機することになる。各バケットは基本的にスイッチ・チップの1つにおいて（但し、異なるポートを通じて）その方向を反転する、すなわち”ターン・アラウンド”するので、これらの全てのバケットにより取られる経路は、集合的に閉ループ・パターン（番号Ⅰ-ⅠⅠ-ⅠⅠⅠ-ⅠⅤで示され、ここでは”サイクル”として参照される）を形成することになる。スイッチ・チップはこれらのどの特定のバケットを最初に経路指定するかを決定できないので、全てのスイッチ・チップは単に待

機し、いずれのバケットも経路指定されない。サイクル内の4つの各々のバケット自身が、残りの3つのバケットを妨害することになる。結果的に、経路指定デッドロックが発生する。このデッドロックが持続する間、対応するパスはバケット・トラフィックを伝搬しない。従って、処理要素 110_a 、 110_{23} 、 110_2 及び 110_{16} は単にバケットの到来を待機し、これらのバケットを要求するアプリケーション部分の処理が延期される。これはすなわち、システム5の処理スループットを低下させることになる。経路指定デッドロックが発生すると、この状態は何らかの手段により解決されるまで無期限に継続する。経路指定デッドロックは比較的稀にしか発生しないが、並列処理システムの規模が増大すると、これらのデッドロックの発生の潜在性も増加する。

【0041】この現象を鑑み、本発明は比較的大規模な大容量並列処理システムにおいて、経路指定デッドロックの発生を防止する手法を提供する。本手法は具体化が非常に単純で高度にコスト有効であり、バケット・ネットワークにおける伝送帯域幅の適度で受諾可能な低減を強要するに過ぎない。

【0042】本手法により、特定の予め定義された経路が、大域経路テーブルの形成の間にこれらの経路を使用する特定の処理要素（ネットワーク・ノード）の相対ロケーションにもとづき考慮から除外される。禁止された経路は、もしそれらが使用されないと、閉ループ経路指定パターン、従って経路指定デッドロックの発生を防止する経路として選択される。経路テーブルに含まれる経路の選択において、システムのある区分、例えばシステムの半分内のノード間をもっぱら流れるバケット・トラフィックを別の区分、例えばシステムの他の半分内のノード間を流れるバケット・トラフィックから分離するために経路が禁止される。この点に関し、システムの共通区分内のノード間を通過するバケットの経路を抽出するために、システムの別の区分を通過するパス（ケーブルなど）を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分内のノード間でバケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。システムの各区分、例えば半分を他の区分、例えばシステムの他の半分内の処理要素対間をもっぱら流れるバケット・トラフィックから分離することにより、これらのバケットの相互作用により生じる経路指定デッドロックが有利に防止される。

【0043】比較的大規模な大容量並列処理システム、例えば512の別々の処理要素を使用するシステムにおいて必要なプロセッサ間経路指定機能を提供するために、システムは多数のスイッチ・ボードを使用する。各スイッチ・ボードは上述されたように同一であり、2つの相互接続ステージ、すなわち個々の処理要素に接続されるノード・スイッチ・ボード（NSB）、及びノード・スイッチ・ボード自身を相互接続するために使用され

る中間スイッチ・ボード（ISB）に編成される。512プロセッサ・システムは、通常、48個の別々のスイッチ・ボードを使用し、これらの内の32個のボードはNSB専用であり、残りの16個のボードはISB専用である。各NSBは16個のそれぞれ異なる処理要素に接続される16ポートと、16個の各ISB上の異なるポートに相互接続される別の16ポートを提供する。この構成では、NSBはバケットを自身が接続される個々の処理要素との間で経路指定し、ISBはバケットを異なるNSB間で経路指定し、全ての完全な経路が、上述のようにバケット・ヘッダに含まれる経路指定バイトにより指定される。

【0044】512プロセッサ・システムの例が、図4にシステム400として示される。図示のように、このシステムは集合的に処理ノード410として示される512の異なる処理要素 415_a 、...、 415_{16} 、...、 415_{496} 、...、 415_{511} を提供し、物理的見地から16個の処理要素を含む32個の物理ラック、特に処理ラック 410_a 、...、 410_{31} に編成される。各ラックはそれぞれのNSBの16ポートに接続される。システム400は32個のNSB 440_a 、 440_1 、 440_2 、 440_3 、 440_4 、 440_5 、...、 440_{30} 及び 440_{31} （NSB0、NSB1などとしても指定される）を含む。各NSBの残りの16ポートは、接続マトリックス450内の個々のケーブルを介して、16個のISB 460 、特にISB 460_a 、 460_1 、 460_2 、...、 460_{16} （ISB0、ISB1などとしても指定される）の各々の対応するポートに相互接続される。例えば、NSB 440 （NSB0）上の16個の各ポートは、16個のISBの対応する異なる1つのポート0に接続されるように示され、それによりNSB 440 は各ISBにバケットを経路指定できる。他のNSBについても図示のように、あらゆるISBに同様に相互接続される。ISBであろうとNSBであろうと、全てのスイッチ・ボードは互いに同一であるが、接続マトリックス450を明瞭に表す都合上、ISBはNSBと異なるように示される。

【0045】システム400において、中間スイッチ・ボードなどの使用に頼る他の大規模大容量並列処理システムと同様、本発明を使用しないと、経路指定デッドロックが発生することが理解される。なぜなら、バケットが異なるNSB間で経路指定される時、図1に示されるシステム5のスイッチ 120_a 及び 120_b 内で、バケット“A”及び“C”がそれらの方向を反転（“ターン・アラウンド”）する時のように、その方向をISB内で反転するからである。図4に示されるように、バケットはISB内においては生成されず、単にそれを通じて別々のNSB間で経路指定されるだけなので、閉ループ経路指定パターンがもしも発生すると、それらはISBに延びる必要があり、NSB内だけに存在するとは限らなくな

る。システム 400 内の経路指定デッドロックは、任意の 1 つまたは複数の NSB 自身だけに制約されない。

【0046】本発明の教示によれば、禁止する経路を決定するために、システム 400 は例証的に半分に区分される。この場合、16 個の連続する NSB（例えば NSB 0 乃至 15、及び NSB 16 乃至 31）、及び 256 個の連続する処理要素（例えばそれぞれ要素 41

50、...、415₂₅₆及び 415₂₅₆、...、415₅₁₁）は各半分に割当てられる。また最初の 8 個の ISB が片方の半分に含まれ、残りの 8 個の ISB が他の半分に含まれる。この点に関し、図 5 を参照すると、図 4 のシステム 400 を構成する全ての NSB 並びに ISB 460 が示される。図示のように、システムは 503 と 507 のそれぞれ半分に区分される。32 個の各 NSB 上のポート 0 などの共通ポート（ラベル付けされていない）が、別々の対応するバス（ケーブル）を介して、単一の ISB 上の 32 個のポートの対応する 1 つに接続される。全ての NSB 上の残りの各ポート及び他の ISB についても同様である。システムの半分 503 は、NSB 440₀ 乃至 440₁₅ 及び ISB 460₀ 乃至 460₁₅ を含む。ここでは NSB 440₀ 乃至 440₁₅ は、バス 510₀₋₀、510₁₋₀、...、510₁₅₋₀ を介して、単一の ISB の 16 個の連続するポート（特にそれらの 3 つだけが示されている）、ここでは ISB 460₀ の特にスイッチ・チップ 530₀ 乃至 530₃ に接続されるように示される。残りのシステムの半分 507 は、NSB 440₁₇ 乃至 440₃₁、及び ISB 460₁₆ 乃至 460₃₁ を含む。同様にこれらの特定の NSB はバス 510₁₆₋₁₆、510₁₇₋₁₆、...、510₃₁₋₁₆ を介して、単一の ISB の対応するポート、ここでは ISB 460₁₆ の特にスイッチ・チップ 540₄ 乃至 540₇ に接続されるように示される。

【0047】システムの共通半分、例えば半分 503 内に配置される処理要素間を通過するパケットに対しては、システムのその半分内に完全に含まれる ISB を含む使用可能な経路（NSB 440₀ と 440₁₅ 間の経路 522、及び NSB 440₃₁ と 440₁₆ 間の経路 524 など）だけが許可され、他の経路（NSB 440₀ と 440₁₆ 間の破線で示される経路 534、及び NSB 440₃₁ と 440₁₆ 間の破線で示される経路 532）は禁止される。従って、後者の任意の経路はシステム初期化の間に、これらの処理要素を接続する大域経路テーブル内に含まれない。禁止された経路はまた、その経路上の”X”により示される。或いはシステムの異なる半分内に配置される処理要素間を通過するパケットに対して、こうした経路が禁止されなくてもよい。この場合、経路選択は大域経路テーブル内に結果的に含むものに対して、システムの半分にもとづく制限無しに、NSB 440₀ と 440₁₆ 間のその時使用可能な全ての経路（1 つの場合もある（特に図示せず））の中から実施される。

【0048】経路の禁止は後述されるように、大域経路テーブルが生成される間に、特定の経路指定指示を処理することにより実行される。この処理は、全ての禁止経路がネットワーク・ノードの所与の対間で定義される経路として選択されることを防止する。

【0049】自身の内部処理要素間で発生するパケット・トラフィックに対応して、システムの各半分を分離し、それにより他の半分に含まれる処理要素間を通過するパケットとの相互作用を排除することにより、経路指定デッドロックが有利に防止される。

【0050】驚くことに、上述のように 512 ポート交換ネットワークの多数の分析の結果、ネットワークを通過する期待されるトラフィック・パターンに関し、本発明の手法によれば、ネットワークの最大伝送帯域幅が支障無い程度に減少するだけであることが判明した。この点に関し、本発明の使用は、ネットワークにおいて使用可能な最大帯域幅のほぼ 74% を確保し、これは期待した約 50% を大きく上回るものである。従って、経路指定デッドロックを回避するための本手法の使用による不利益は、特に獲得される利点を鑑みれば、極めて受諾可能と言える。

【0051】上述の説明を鑑み、図 6 は、図 4 に示されるシステム 400 内に配置されるサービス・プロセッサ（例として処理要素 415₅₁₁）内で実行される、本発明の教示によりパケット経路を定義する経路テーブル発生器ルーチン 600 のハイレベル流れ図を示す。ルーチン 600 は、上述のようにサービス・プロセッサ内で実行される初期化ルーチンの 1 部である。

【0052】図 6 に示されるように、ルーチン 600 へのエントリに際し、実行は最初にブロック 610 に移行し、トポロジ・ファイル及び付随する経路指定指示を読み出す。デッドロック回避経路指定を提供するために、パケット・ネットワーク内の各装置、例えばスイッチ回路（または特にそこで使用されるスイッチ・チップ）に対応する適切な指示、すなわちその装置を通過する経路指定が制限されているか否か、換言すると、パケットがこの回路を通じて方向を反転可能か否かを示す指示が、トポロジ・ファイル内に含まれなければならない。ネットワーク 100 を実現する図 1 に示されるスイッチ・ボードについて考えてみる。上述のように、各スイッチ・チップのポート 0 乃至 3 は、スイッチ・ボードの外部のリンクに接続され、各スイッチ・チップのポート 4 乃至 7 は接続マトリックス 140 内のリンク（ケーブル）に接続され、それを通じて、同一ボード内の別のスイッチ・チップのポートに接続される。トポロジ・ファイル内において特定のスイッチ・チップに対して指定される経路指定指示”nr”は、そのチップに関し、経路指定制限が存在しないことを意味する。パケットはこのチップ上の任意の 8 個のポートに入力することができ、チップ上の他のポートから去ることができる。この例では、パケッ

トはチップ内でその方向を反転（“ターン・アラウンド”）することができる。或いはスイッチ・チップに対して、トポロジ・ファイル内に経路指定指示“n-i-t”が存在すると、ポート 4 乃至 7 に入力するバケットは、チップ上のポート 0 乃至 3 からだけ出力するようにその経路指定が制限される。すなわち、そのバケットはチップ内で方向を反転することを禁止される。しかしながら、“n-i-t”指示は、スイッチ・チップの任意のポート 0 乃至 3 に到来するバケットに対しては制限せず、これはそのチップ上の任意の他のポートに経路指定される。デッドロック回避指示を有するトポロジ・ファイル内のサンプル行は次の通りである。

【数 1】aux routing n-i-t 330 331 332 333

ここで、“aux routing”は、経路指定指示を有する補助行を意味する。そして、“330 331 332 333”は、トポロジ・ファイル内で使用されるフォーマットの特定のスイッチ回路の数値識別子である。

【0053】ブロック 610 が完全に実行されると、実行はブロック 620 に移行し、トポロジ・ファイル内で指定される各ケーブル（リンク）に関連する重みを 0 に設定する。更に出所ノード・カウンタに相当するノード i が 0 に初期化される。その後、実行はブロック 630 に移行する。この特定のブロックはトポロジ・ファイル内に含まれるデータと付随するデッドロック回避経路指定指示とを一緒に使用することにより、バケット・ネットワークを通じて現出所ノード（ノード i）をシステム内のあらゆる宛先ノードに接続するために使用可能な経路のセットを抽出する。具体的には、既知のブレッズ・ファースト探索（breadth-first search）により、最短長を有する経路、すなわち必ずしも物理的に最短長を有するわけではないが、最少の個々のリンク（ケーブル）を有する経路が選択される。上述された各スイッチ回路に関連するデッドロック回避経路選択を表す擬似コードを次に示す。

【数 2】case of routing_directive is

```
{
  "nr":total_permmissible_oports=8;
  /*スイッチ・チップ上の全出力ポート*/
  "n-i-t":if(input_port<4)
    total_permmissible_oports=8;
  else
    total_permmissible_oports=4;
}
i=0;
while(i<total_permmissible_oports) do
{
  permmissible_oport[i]=i
  i=i+1
}
```

【0054】最短パス経路のセットの選択は、ブロック

640 に示されるように、経路が現出所ノードから全ての宛先ノードに延びるまで、すなわち経路が宛先ベースになるまで出所ベースで発生する。1つの最短長経路だけが出所ノードから宛先ノードに生じる場合、その経路が選択されて使用される。或いは複数のこうした経路がこの出所ノードと共通宛先ノード間で生じる場合、集合的に最低の重みのケーブルを有する経路が選択される。重みベースの選択により、バケット・ネットワーク全体を通じて最小のケーブルの共用を維持するように、トラフィック負荷が平衡される。出所ノードと宛先ノードの間で特性の経路が選択されると、その経路内の各ケーブルに関連する重みが 1 だけ増分される。ブロック 630 及び 640 は、理解を容易にするために別個のブロックとして示されるが、オペレーションは一般に結合される。

【0055】全ての宛先ノードに対応して全ての経路が選択されると、実行はブロック 650 に移行し、全ての選択経路を大域経路テーブルに書込む。これにより現出所ノードに対する経路テーブルが形成される。その後、実行は判断ブロック 660 に移行し、ネットワーク内のあらゆるノードに対して、経路テーブルが大域経路テーブルに書込まれたかどうかを判断する。経路テーブルがあらゆるノードに対して書込まれていない場合には、判断ブロック 660 は実行を否定パス 667 を介して、ブロック 670 に移行させる。この後者のブロックの実行により、出所ノード・カウンタ i が 1 増分される。実行は次にパス 675 を介して、ブロック 630 ヘルプして戻り、次に続くノードの経路を判断しそれを書込む。或いは経路テーブルが全てのノードについて書込まれると、実行は判断ブロック 660 からの肯定パス 663 を介して、ルーチン 600 を終了する。このルーチンの実行の後、初期化処理の完了に先立ち、上述のサービス・プロセッサはネットワークを通じ、大域経路テーブルの対応部分を、自身を含む各々の及びあらゆる個々の処理要素に提供する（特にコピーする）。そして、こうして記憶されたものが、後に局所経路テーブルとして使用される。この部分は、その特定の処理要素が出所ノードの時に選択される経路を含むだけである。

【0056】これまでの説明から当業者には理解されるように、本発明は 512 の別々の処理要素を有する大容量並列処理システムに関連して述べられてきたが、もちろんこれに限るものではない。実際に、本発明は実質的に双方向マルチステージ相互接続クロスポイント・ベース・バケット・ネットワークを使用する任意のサイズの並列処理システムにおける、経路指定デッドロックの回避にも適用される。その点に関し、本発明は 64 プロセッサ・システム、256 プロセッサ・システム、及び他のサイズの類似のシステム、並びにマルチステージ相互接続クロスポイント・バケット・ネットワークを使用する他のシステムにそれらの最終利用に関係無く、容易に

組込むことが可能である。

【0057】更に、本発明の教示はバケット・ネットワークを2つの別々の半分に区分し、それらの間の経路指定を制限する状況において述べられたが、こうしたネットワークは本発明により任意の数の別々の区分に分割され、これらの各区分内だけをもつばら通過するバケット・トラフィックを分離するように編成される。もちろん、区分数が増加すると、それに伴い区分化を達成するために必要となる禁止経路の数も増加する。残念ながら、禁止経路の数が増えるとバケット・トラフィックを伝搬する使用可能な経路が減少し、従って、ネットワークの伝送帯域幅が減少する。支障の無い帯域幅の減少を鑑みると、達成されるバケット分離及びデッドロック回避の点から、2つの区分が優れたトレードオフを提供することが判明した。

【0058】まとめとして、本発明の構成に関して以下の事項を開示する。

【0059】(1) バケット・ネットワークの外部の複数のノードを集散的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有する装置において、バケットが前記ネットワーク及び少なくとも1つの前記スイッチを介して、規定経路上を第1の前記ノードから第2の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する実質的な方法であって、バケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を第1に定義するステップであって、前記の各定義経路が少なくとも1つのリンクに伸び、第1のネットワーク区分だけに接続される第1及び第2の前記ノード間を通過するバケットが、第2のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第1及び前記第2のネットワーク区分に分割するように前記規定経路を定義する、前記第1の定義ステップと、全ての前記規定経路を結果の経路テーブルに記憶するステップとを含む、方法。

(2) 前記第1の定義ステップが、前記第1及び前記第2のネットワーク区分にそれぞれ接続される第3及び第4のノード間を通過するバケットが、前記第1及び前記第2のネットワーク区分間に伸びる少なくとも1つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する第2の定義ステップを含む、前記

(1) 記載の方法。

(3) 前記ネットワーク内において出所ノードから宛先ノードに経路指定されるバケットをアセンブルする際に、前記第3及び前記第4の両ノード内において、前記バケットの結果的経路を生成するために、前記経路テーブルをアクセスするステップと、前記結果的経路を前記バケットにコピーするステップと、前記バケットを前記

結果的経路上で前記ネットワークを介して経路指定するステップとを含む、前記(2)記載の方法。

(4) 前記経路テーブルの異なる部分を、前記複数の各ノードに対応する別々の局所経路テーブルにダウンロードするステップであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記ダウンロード・ステップを含み、前記経路コピー・ステップが、前記出所ノードから前記宛先ノードに伝搬されるバケットの前記結果的経路を生成するために、前記バケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスするステップを含む、前記(3)記載の方法。

(5) 前記の各バケットが少なくとも1つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各バケットが前記ネットワークを伝わる経路を集散的に指定し、各個々の前記経路バイトが前記各バケットが対応する前記クロスポイント・スイッチの1つを横断する経路を定義し、前記結果的経路のコピー・ステップが前記結果的経路内の各連続する前記経路バイトの値を前記ヘッダ内の別々の対応する連続経路バイトにコピーするステップを含む、前記(4)記載の方法。

(6) 各ネットワーク区分が前記ネットワークの異なる半분을構成する、前記(5)記載の方法。

(7) 前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第1の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記バケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、前記(5)記載の方法。

(8) 前記第1の定義ステップが、トポロジ・ファイル内のネットワーク装置及び相互接続データにตอบสนองして、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの1つへの全ての使用可能な最短パス経路を決定するステップであって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定ステップと、ある前記出所ノードとある前記宛先ノード間で1つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込むステップと、ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集散的に最小の重みを有する1つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択するステップと、前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分するステップとを含む、前記(5)記載

の方法。

(9) 前記全ての使用可能な経路の決定ステップが、前記全ての使用可能な最短パス経路を突き止めるブレード・ファースト探索を実行するステップを含む、前記

(8) 記載の方法。

(10) 前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第1の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記パケット

経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、前記(9)記載の方法。

(11) 各ネットワーク区分が前記ネットワークの異なる半分を構成する、前記(10)記載の方法。

(12) パケット・ネットワークの外部の複数のノードを集散的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有するシステムにおいて、パケットが前記ネットワーク及び少なくとも1つの前記スイッチを介して、規定経路上を第1の前記ノードから第2の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する装置であって、パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を定義する第1の手段であって、前記の各定義経路が少なくとも1つのリンクに伸び、第1のネットワーク区分だけに接続される第1及び第2の前記ノード間を通過するパケットが、第2のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第1及び前記第2のネットワーク区分に分割するように前記規定経路を定義する、前記第1の定義手段と、全ての前記規定経路を結果の経路テーブルに記憶する手段とを含む装置。

(13) 前記第1の定義手段が、前記第1及び前記第2のネットワーク区分にそれぞれ接続される第3及び第4のノード間を通過するパケットが、前記第1及び前記第2のネットワーク区分間に伸びる少なくとも1つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する、前記(12)記載の装置。

(14) 前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする間に、前記第3及び前記第4の両ノード内において、前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスし、前記結果的経路を前記パケットにコピーし、前記パケットを前記結果的経路上で前記ネットワークを介して経路指定する手段を含む、前記(13)記載の装置。

(15) 前記経路テーブルの異なる部分がダウンロードされる前記複数の各ノードに対応する別々の局所経路テ

ーブルであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記局所経路テーブルと、前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスする手段とを含む、前記(14)記載の装置。

(16) 前記の各パケットが少なくとも1つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集散的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの1つを横断する経路を定義し、前記結果的経路内の各連続する前記経路バイトの値を、前記ヘッダ内の別々の対応する連続経路バイトにコピーする、前記(15)記載の装置。

(17) 各ネットワーク区分が前記ネットワークの異なる半分を構成する、前記(16)記載の装置。

(18) 前記第1の定義手段が、トポロジ・ファイル内のネットワーク装置及び相互接続データにตอบสนองして、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの1つへの全ての使用可能な最短パス経路を決定する手段であって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定手段と、ある前記出所ノードとある前記宛先ノード間で1つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込む手段と、ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集散的に最小の重みを有する1つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択する手段と、前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分する手段とを含む、前記(16)記載の装置。

(19) 前記システムが並列処理システムであり、前記の各ノードが別々の処理要素を含む、前記(16)記載の装置。

(20) 前記並列処理システムが512個の別々の処理要素を含み、前記スイッチが32ポート・スイッチ・ボードに編成され、前記システムが32個のノード・スイッチ・ボード(NSB)と16個の中間スイッチ・ボード(ISB)による複数のスイッチ・ボードを含み、前記ISBが、前記の各NSB上の16ポートのそれぞれが、異なる対応するリンクを介して、前記の各NSB上の同一の対応するポートに接続されるように、また前記の各NSB上の残りの16ポートが16個の異なる連続する前記処理要素に接続されるように、全ての前記NS

Bを集合的に相互接続する、前記(19)記載の装置。
【0060】

【発明の効果】以上説明したように、本発明によれば、大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおいて、デッドロックの無い経路指定を確立する単純でコスト・パフォーマンスの良い装置及び方法が提供される。

【図面の簡単な説明】

【図1】 32個の別々の処理要素を使用する従来の並列処理システム5のハイレベル・ブロック図である。

【図2】 図1に示されるシステム5を通過するパケット300及びその構成フィールドを表す図である。

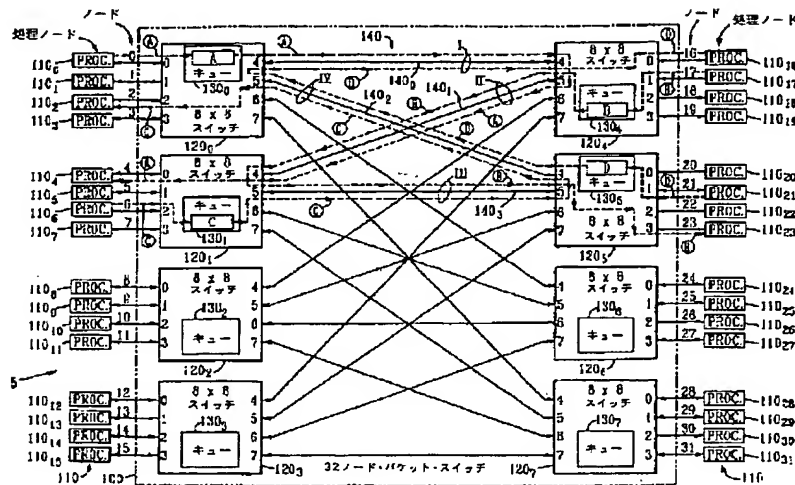
【図3】 図1に示されるシステム5を構成する処理ノード110、及び特にこれらのノードのメモリ内に存在してパケット経路指定を達成する様々なファイル及びテーブルを示す図である。

【図4】 512の処理要素を含み、本発明の教示を使用する並列処理システム400のハイレベル・ブロック図である。

【図5】 システム400内に配置される中間スイッチ・ボード(ISB)及びそれらの相互接続ノード・スイッチ・ボード(NSB)を示し、パケット経路の例が本発明の教示により決定される。

【図6】 サービス・プロセッサ内で実行される経路テ

【図1】

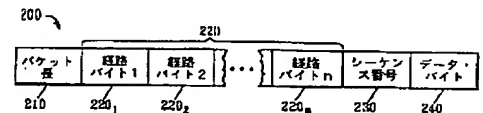


ブル発生器ルーチン600のハイレベル流れ図であり、図4に示される処理要素415₅₁₁が、本発明の教示によりパケット経路を定義するためにシステム400内に配置される。

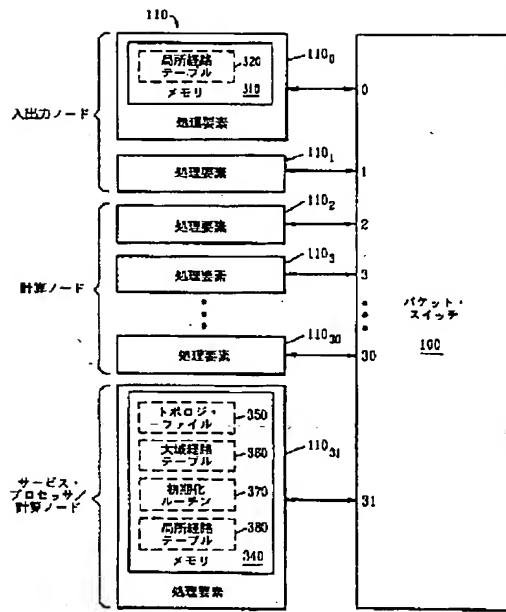
【符号の説明】

- 5、400 32プロセッサ並列処理システム
- 100 ノード・パケット・スイッチ
- 110 処理要素、処理ノード
- 120 8×8双方向スイッチ回路
- 140、450 接続マトリックス
- 200 パケット
- 210 長さフィールド
- 220 経路指定フィールド
- 230 シーケンス番号フィールド
- 240 データ・フィールド
- 320、380 局所経路テーブル
- 340 メモリ
- 350 トポロジ・ファイル
- 360 大域経路テーブル
- 370 初期化ルーチン
- 460 IBS
- 600 経路テーブル発生器ルーチン
- 667 否定パス

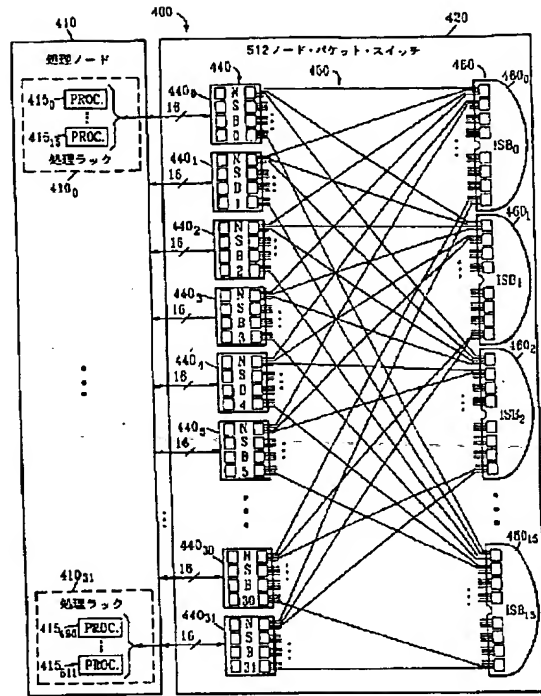
【図2】



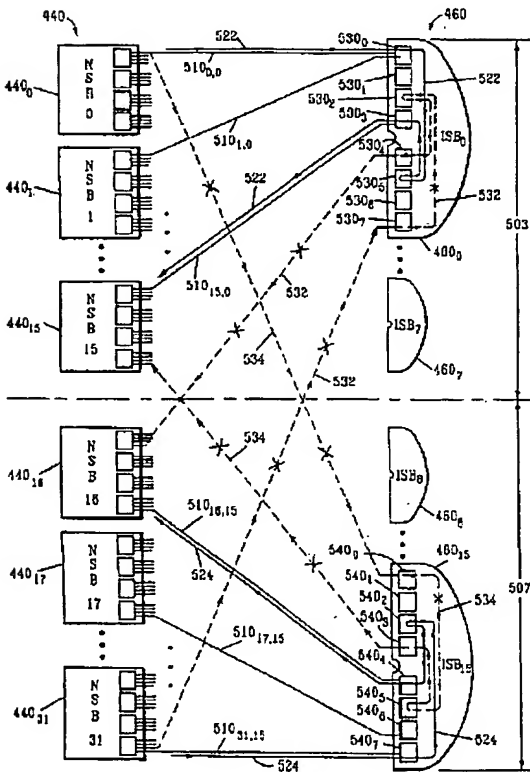
【図 3】



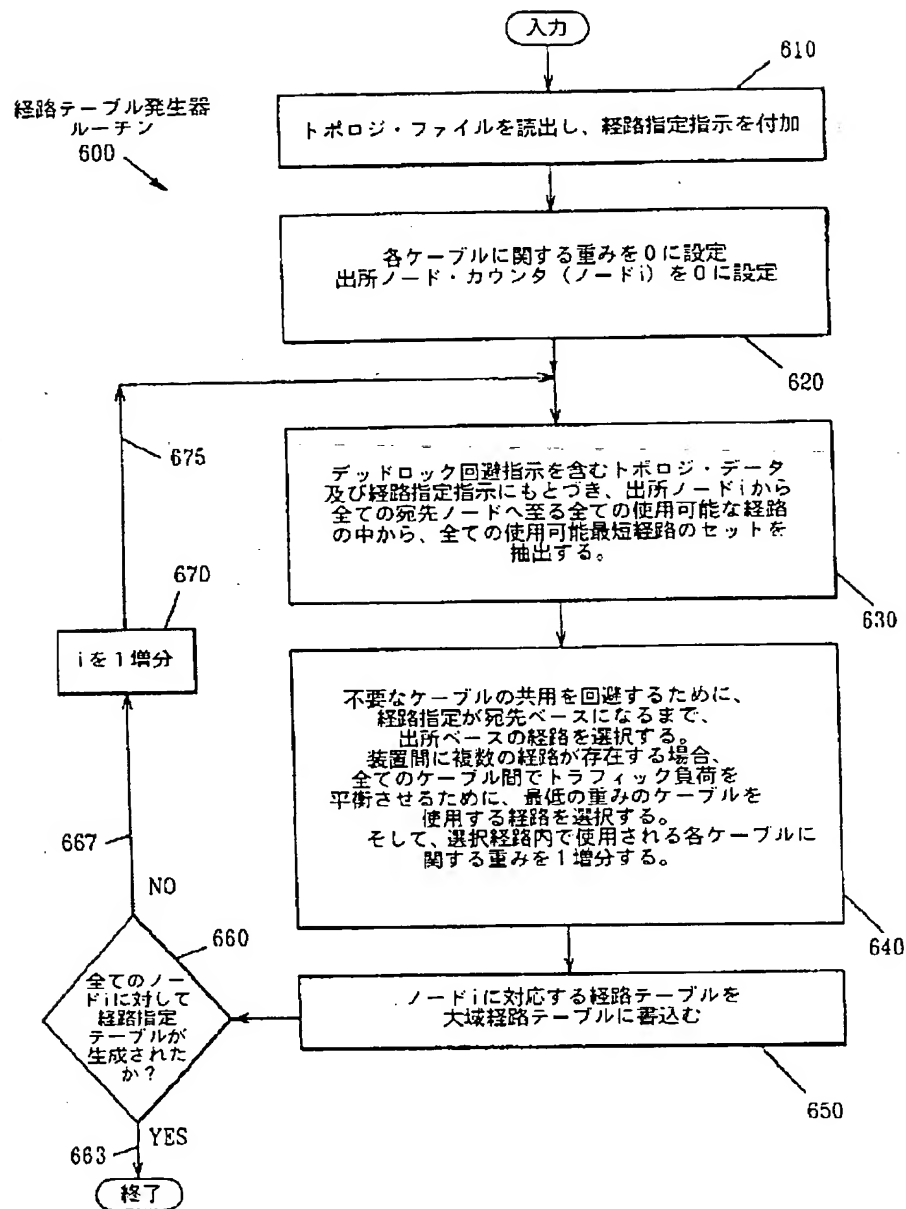
【図 4】



【図 5】



【図6】



フロントページの続き

(72)発明者 ロバート・フレデリック・スタック
アメリカ合衆国12477、ニューヨーク州ソ
ガティーズ、リッジ・ロード 14

(72)発明者 クレイグ・ブライアン・スタンゲル
アメリカ合衆国06801、コネチカット州ベ
スル、グリーン・パスチャー・ロード 10

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ **BLACK BORDERS**
- ☒ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.